CLAIMS

1. A rasterizer interpolator comprising:

a setup unit for calculating pixel coverage of a graphics primitive against one or more tiles;

a plurality of graphics pipelines, wherein each pipeline is dedicated to one of a plurality of portions of an output screen with said setup unit distributing instructions to said pipelines, each pipeline further comprises:

a rasterizer configured to receive said primitive determined to result in the generation of covered pixels in tiles in a screen region dedicated to said pipeline, and perform one or more graphics processing operations on said tiles.

2. The rasterizer interpolator of claim 1 wherein said rasterizer further includes:

a scan converter and a Hierarchical-Z unit configured to perform a coarse grain tiling process on said tiles.

3. The rasterizer interpolator of claim 2 wherein said scan converter computes a list of tiles in said pipeline covered by said graphics primitive.

4. The rasterizer interpolator of claim 2 wherein said Hierarchical-Z unit works in conjunction with the data from said scan converter to compute a list of visible tiles in said pipeline covered by said graphics primitive.

5. The rasterizer interpolator of claim 1 wherein said setup unit uses polygon vertex data in said primitive to calculate pixel coverage.

6. The rasterizer interpolator of claim 5 wherein said calculation performed by said setup unit dynamically takes into account different configurations of said plurality of graphics pipelines.

7. The rasterizer interpolator of claim 1 wherein the size of said tiles is configurable.

8. The rasterizer interpolator of claim 1 wherein the width and height of said tiles are the same.

9. The rasterizer interpolator of claim 1 wherein the number of said plurality of screen portions is 2^n.

10. The rasterizer interpolator of claim 1 wherein said setup unit distributes instructions to said pipelines in a non-contagious manner.

11. A rasterizer interpolator comprising:

a setup unit for calculating pixel coverage of a graphics primitive against one or more tiles;

a plurality of graphics chips, wherein each chip is dedicated to one of a plurality of portions of an output screen with said setup unit distributing instructions to said chips, each chip further comprises:

one or more parallel pipelines configured to receive said primitive determined to result in the generation of covered pixels in tiles a screen region dedicated to said pipeline and perform one or more graphics processing operations on said tiles.

12. The rasterizer interpolator of claim 11 wherein said setup unit performs super tiling in distributing said tiles to said plurality of graphics chip.

13. The rasterizer interpolator of claim 11 wherein said graphics processing operations include coarse grain tiling.

14. The rasterizer interpolator of claim 11 further comprising:

a cache in each of said graphics chips configured to store said tiles.

15. A method for interpolating tiles to a rasterizer comprising:

determining pixel coverage of a plurality of graphics primitives against one or more tiles;

distributing said primitives to one or more pipelines, wherein each of said pipelines is granted ownership to each of said tiles that fall into a designated screen region;

using a rasterizer in each of said pipeline to receive said primitives and perform one or more graphics processing operations on said tiles.

16. The method of claim 15 wherein said determining uses polygon vertex data in said primitives to calculate pixel coverage.

17. The method of claim 15 wherein said determining dynamically takes into account different configurations of said plurality of graphics pipelines.

18. The method of claim 15 wherein the size of said tiles is configurable.

19. The method of claim 15 wherein the width and height of said tiles are the same.

20. The method of claim 15 wherein the number of said plurality of screen portions is 2^n.

21. The method of claim 15 further comprising:

using a scan converter and a Hierarchical-Z unit to perform a coarse grain tiling process on said tiles.

22. The method of claim 21 wherein said step of using further comprises:

computing a list of tiles in a current pipeline and covered by a current graphics primitive;

generating a first mask value specifying which of intermediate tiles within each of said tiles on said list are visible;

generating a sub-list containing smaller quad tiles within said intermediate tiles;

computing a second mask value specifying which of said quad tiles are visible and a z plane equation.

23. The method of claim 22 wherein said tiles are of size 8x8.

24. The method of claim 22 wherein said intermediate tiles are of size 4x4.

25. The method of claim 22 wherein said quad tiles are of size 2x2.

26. The method of claim 15 wherein said ownership of said tiles is assigned to said pipelines in a non-contagious manner.

27. A method for interpolating tiles to a rasterizer comprising:

determining pixel coverage of a plurality of graphics primitives against one or more tiles;

distributing said primitives to one or more graphic chips, wherein each of said graphics chip is granted ownership to each of said tiles that fall into a designated screen region;

using one or more pipelines in each of said graphics chip to receive said primitives, wherein each pipeline performs one or more graphics processing operations on said tiles.

28. The method of claim 27 wherein said distributing uses a super tiling process.

29. The method of claim 27 wherein said graphics processing operations include coarse grain tiling.

30. The method of claim 27 further comprising:

storing each of said tiles in a cache in each of said graphics chips.

31. A computer program product comprising:

a computer usable medium having computer readable program code embodied therein configured to interpolate tiles to a rasterizer, said computer program product comprising:

computer readable code configured to cause a computer to determine pixel coverage of a plurality of graphics primitives against one or more tiles;

computer readable code configured to cause a computer to distribute said primitives to one or more pipelines, wherein each of said pipelines is granted ownership to each of said tiles that fall into a designated screen region;

computer readable code configured to cause a computer to use a rasterizer in each of said pipeline to receive said primitives and perform one or more graphics processing operations on said tiles.

32. The computer program product of claim 31 wherein said computer readable code configured to cause a computer to determine uses polygon vertex data in said primitives to calculate pixel coverage.

33. The computer program product of claim 31 wherein said computer readable code configured to cause a computer to determine dynamically takes into account different configurations of said plurality of graphics pipelines.

34. The computer program product of claim 31 wherein the size of said tiles is configurable.

35. The computer program product of claim 31 wherein the width and height of said tiles are the same.

36. The computer program product of claim 31 wherein the number of said plurality of screen portions is 2^n.

37. The computer program product of claim 31 further comprising:

computer readable code configured to cause a computer to use a scan converter and a Hierarchical-Z unit to perform a coarse grain tiling process on said tiles.

38. The computer program product of claim 37 wherein said computer readable code configured to cause a computer to use further comprises:

computer readable code configured to cause a computer to compute a list of tiles in a current pipeline and covered by a current graphics primitive;

computer readable code configured to cause a computer to generate a first mask value specifying which of intermediate tiles within each of said tiles on said list are visible;

computer readable code configured to cause a computer to generate a sub-list containing smaller quad tiles within said intermediate tiles;

computer readable code configured to cause a computer to compute a second mask value specifying which of said quad tiles are visible and a z plane equation.

39. The computer program product of claim 38 wherein said tiles are of size 8x8.

40. The computer program product of claim 38 wherein said intermediate tiles are of size 4x4.

41. The computer program product of claim 38 wherein said quad tiles are of size 2x2.

42. The computer program product of claim 31 wherein ownership of said tiles is assigned to said pipelines in a non-contagious manner.

43. A computer program product comprising:

a computer usable medium having computer readable program code embodied therein configured to interpolate tiles to a rasterizer, said computer program product comprising:

computer readable code configured to cause a computer to determine pixel coverage of a plurality of graphics primitives against one or more tiles;

computer readable code configured to cause a computer to distribute said primitives to one or more graphic chips, wherein each of said graphics chip is granted ownership to each of said tiles that fall into a designated screen region;

computer readable code configured to cause a computer to use one or more pipelines in each of said graphics chip to receive said primitives, wherein each pipeline performs one or more graphics processing operations on said tiles.

44. The computer program product of claim 43 wherein said computer readable code configured to cause a computer to distribute uses a super tiling process.

45. The computer program product of claim 43 wherein said graphics processing operations include coarse grain tiling.

46. The computer program product of claim 43 further comprising:

computer readable code configured to cause a computer to storing each of said tiles in a cache in each of said graphics chips.